STATUS OF THE CLAIMS

The status of the claims of the present application stands as follows:

1. - 8. **(Canceled)**

9. **(Previously presented)** A method of building fault tolerant logic functions in an integrated circuit, comprising the steps of:

creating an integrated circuit design description using a hardware design language at the register-transfer level;

adding a fault tolerant operator to the particular logic functions in said integrated circuit design description; and

building redundant copies for the particular logic functions having a fault tolerant operator.

10. **(Previously presented)** A method according to claim 9, wherein said integrated circuit design description in said creating step is for a FPGA.

11. **(Previously presented)** A method according to claim 9, wherein said building step includes building at least three physical copies of each logic function having a fault tolerant operator.

12. **(Previously presented)** A method according to claim 11, further comprising the step of determining which of said at least three physical copies is faulty.

13. **(Previously presented)** A method according to claim 12, wherein said step of determining includes using a majority voter.

14. **(Previously presented)** A method according to claim 13, further comprising the step of receiving at said majority voter an output value from each of said at least three physical copies of each logic function, wherein for any minority output value at said majority voter, said respective copy is deemed faulty.

15. **(Previously presented)** A method according to claim 12, further comprising the step of disabling any of said at least three physical copies that are faulty.

16. **(Previously presented)** A method of creating fault tolerant logic functions during design of an integrated circuit using a HDL at the RTL and a logic synthesis tool, comprising the steps of:

creating an integrated circuit design description using the hardware design language at the RTL;

adding a fault tolerant operator to each logic function in said integrated circuit design description;

processing said integrated circuit design description through the logic synthesis tool after said adding step; and

including a fault redundant scheme in said integrated circuit design description for each logic function having a fault tolerant operator.

17. **(Previously presented)** A method according to claim 16, wherein said fault redundant scheme includes building at least three physical copies of each logic function having a fault tolerant operator.

18. **(Previously presented)** A method according to claim 17, further comprising the step of determining which of said at least three physical copies is faulty.

19. **(Previously presented)** A method according to claim 18, wherein said step of determining includes using a majority voter.

20. **(Previously presented)** A method according to claim 19, further comprising the step of receiving at said majority voter an output value from each of said at least three physical copies of each logic function, wherein for any minority output value at said majority voter, said respective copy is deemed faulty.

21. **(Previously presented)** A method according to claim 18, further comprising the step of disabling any of said at least three physical copies that are faulty.

22. **(Previously presented)** A method according to claim 16, wherein said integrated circuit design description in said creating step is for a FPGA.

23. **(Previously presented)** A system for creating fault tolerant logic functions during design of an integrated circuit using a HDL at the RTL and a logic synthesis tool, the system comprising:

means for creating an integrated circuit design description using the hardware design language at the RTL;

means for adding a fault tolerant operator to each logic function in said integrated circuit design description;

means for synthesizing said integrated circuit design description after said means for adding step; and

means for building fault redundancy into said integrated circuit design description for each logic function having a fault tolerant operator.

24. **(Previously presented)** A system according to claim 23, wherein said means for building fault redundancy includes building at least three physical copies of each logic function having a fault tolerant operator.

25. **(Previously presented)** A system according to claim 24, further comprising the step of determining which of said at least three physical copies is faulty.

26. **(Previously presented)** A system according to claim 25, wherein said step of determining includes using a majority voter.

27. **(Previously presented)** A system according to claim 26, further comprising the step of receiving at said majority voter an output value from each of said at least three physical copies of each logic function, wherein for any minority output value at said majority voter, said respective copy is deemed faulty.

28. **(Previously presented)** A system according to claim 23, wherein said integrated circuit design description in said creating step is for a FPGA.

[THE REST OF THIS PAGE INTENTIONALLY LEFT BLANK]